Assignment #1

An Exercise of SQL Using SQL*Plus

KAIST

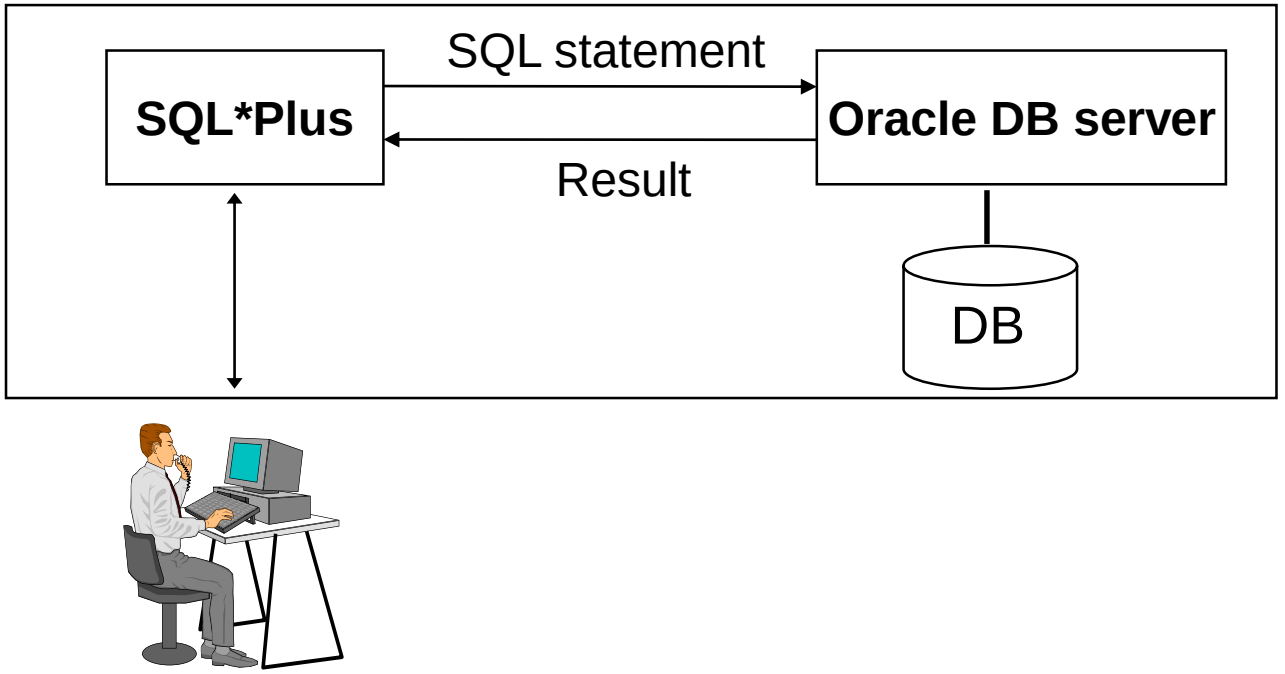
Myoung Ho Kim

Table of Contents

SQL*Plus

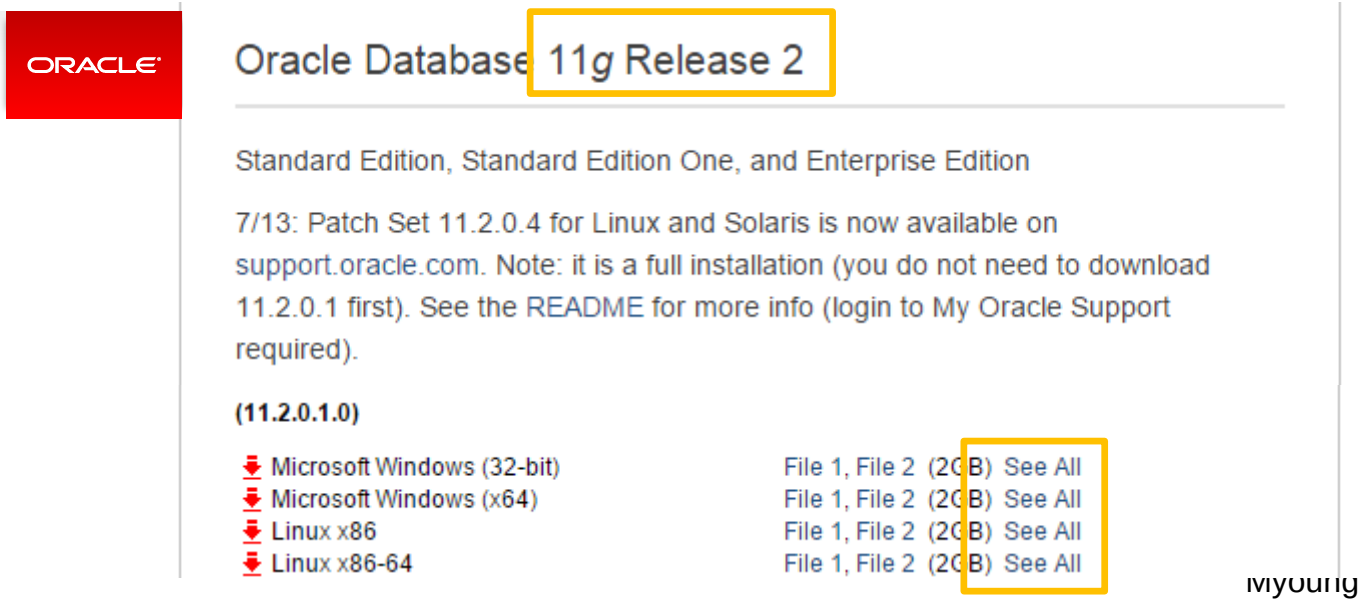
Oracle SQL*Plus

- An Oracle command-line utility program that can run SQL commands interactively or from a script.



Access to DB using SQL*Plus

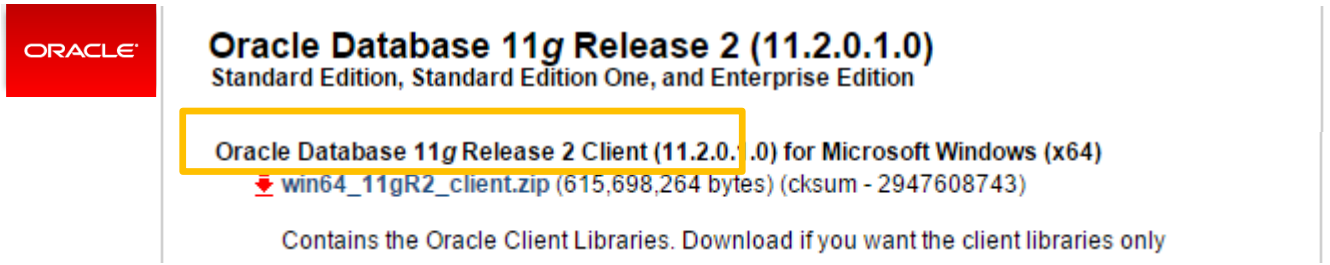
- **Install Oracle Client**
  1) http://www.oracle.com/technetwork/database/enterprise-edition/downloads/index.html
  2) Scroll down to *Oracle Database 11g Release 2*
  3) Click *See All*



Access to DB using SQL*Plus

- **Install Oracle Client**

  4) Download *Oracle Database 11g Release 2 Client*

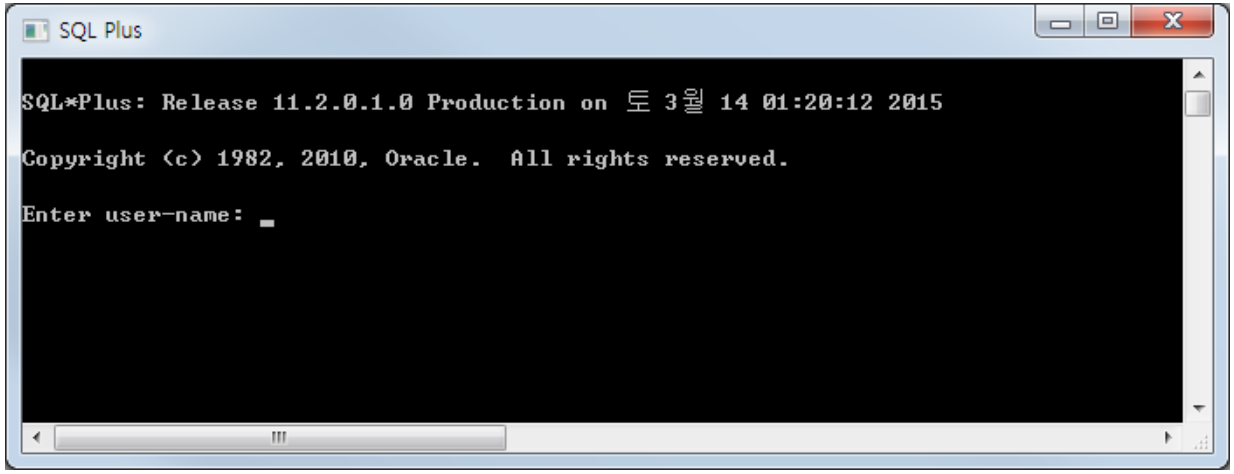


  5) Run *setup.exe*

  6) Check 'Manager' type

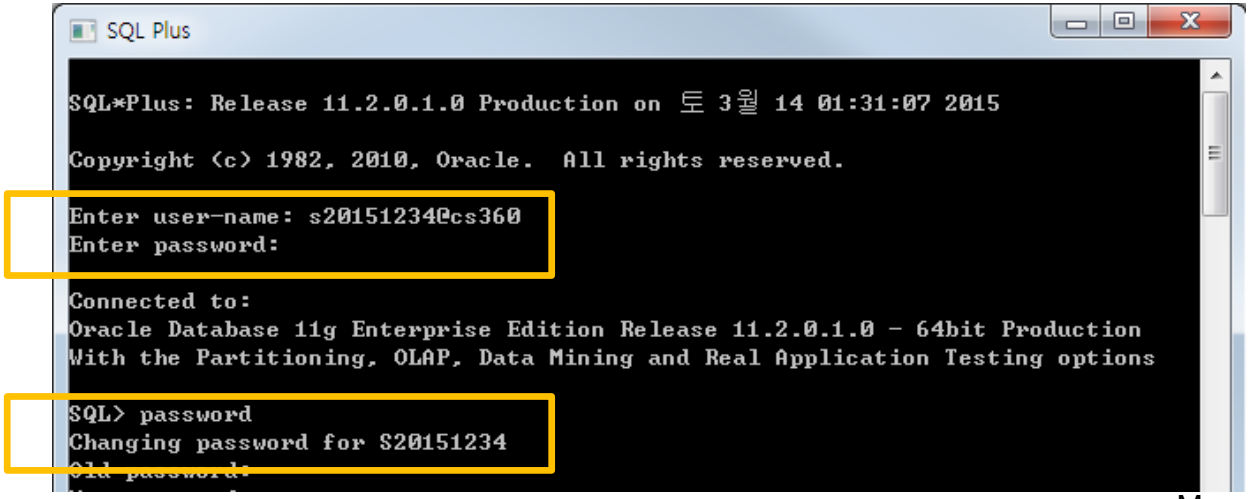
Access to DB using SQL*Plus (cont'd)

- **Run SQL Plus**
  1) Download *tnsnames.ora* from course homepage and copy it to (directory that *Oracle Client* is installed)₩network₩admin
  2) Run *SQL Plus*



Access to DB using SQL*Plus (cont'd)

- **Access to database**
  - User-name: s[studentID]@cs360
    - » ex) If your studentID is 20151234, then your user-name is *s20151234@cs360*
  - Password: s[studentID]
    - » ex) If your studentID is 20151234, then your password is *s20151234*
      - It is recommended to change your password for security



SQL*Plus Commands

SQL*Plus Commands

- **SQL*Plus buffer commands**
  - **LIST** List one or more lines of the SQL buffer
  - **CHANGE** Change text on the current line in the buffer
  - **DEL** Delete one or more lines of the buffer
  - **APPEND** Add specified text to the end of the current line in the buffer
  - **RUN** Execute the SQL command currently stored in the SQL buffer
  - **CLEAN BUFFER** Erase the SQL command currently stored in the SQL buffer
- **SQL*Plus file commands**
  - **SAVE** Save the contents of the SQL buffer in a host operating system file
  - **GET** Load a host operating system file into the SQL buffer
  - **START** Execute the contents of the specified script
  - **SPOOL** Store query results in an operating system file (.sql)
  - **HOST** Execute a host operating system command without leaving SQL*Plus
  - **EDIT** Open a text editor like the notepad to edit an text file (.sql, .lst, etc)

SQL*Plus Commands (cont'd)

- **SQL*Plus buffer commands**
  - » Besides sending SQL statements to the server, SQL*Plus also saves them into a local buffer and allow users to view and change the statements
  - – ***LIST***
    - » List one or more lines of the SQL buffer
  - – ***CHANGE***
    - » Change text on the current line in the buffer
  - – ***RUN(***or ***/)***
    - » Execute the SQL command currently stored in the SQL buffer

SQL*Plus Commands (cont'd)

- **SQL*Plus buffer commands**
  - ***DEL***
    - » Delete one or more lines of the buffer
  - ***APPEND/ INPUT***
    - » Add specified text / line(s) to the end of the current line in the buffer
  - ***CLEAR BUFFER***
    - » Erase the SQL command currently stored in the SQL buffer

SQL*Plus Commands (cont'd)

- **SQL*Plus buffer commands**
  - ***LIST, CHANGE***

```
SQL> select customer_numberr
  2 , email
  3 from customer
  4 where state='TX';
select customer_numberr
       *
ERROR at line 1:
ORA-00904: "customer_numberr": invalid
identifier

SQL> list;
  1  select customer_numberr
  2  , email
  3  from customer
  4* where state='TX'
```

Show the contents in the sql buffer

```
SQL> list 1;
  1* select customer_numberr

SQL> change /numberr/number;
  1* select customer_number

SQL> list;
  1  select customer_number
  2  , email
  3  from customer
  4* where state='TX'
```

List the first line

Change text on the current line

Show the contents in the sql buffer

SQL*Plus Commands (cont'd)

- **SQL*Plus buffer commands**
  - ***RUN(*or */), DEL***

```
SQL> list;
  1  select customer_number
  2  , email
  3  from customer
  4* where state='TX'

SQL> /

CUSTOMER_NUMBER      EMAIL
-------------------------  -------------------------
321654987                  bfarmer@email.com
```

Show the contents in the sql buffer

Execute the command currently stored in the sql buffer

```
SQL> list;
  1  select customer_numberr
  2  , email
  3  from customer
  4* where state='TX'

SQL> del 4;

SQL> list;
  1  select customer_number
  2  , email
  3* from customer
```

Show the contents in the sql buffer

Delete the 4th line

Show the contents in the sql buffer

SQL*Plus Commands (cont'd)

- **SQL*Plus buffer commands**
  - ***APPEND***

```
SQL> list;
  1  select customer_numberr
  2  , email
  3  from customer
  4* where state='TX'
```

Show the contents in the sql buffer

```
SQL> list 2;
  2* , email
```

Show the second line

```
SQL> append ,city
  2* , email,city
```

Add text to the end of the current line in the buffer

```
SQL> list;
  1  select customer_number
  2  , email,city
  3  from customer
  4* where state='TX'
```

Show the contents in the sql buffer

```
SQL> /
CUSTOMER_NUMBER      EMAIL                CITY
-------------------- -------------------- --------------------
321654987            bfarmer@email.com    DALLAS
```

Execute the command currently stored in the sql buffer

SQL*Plus Commands (cont'd)

- **SQL*Plus buffer commands**
  - ***INPUT, CLEAR BUFFER***

```
SQL> list;
  1  select customer_number
  2  , email,city
  3  from customer
  4* where state='TX'

SQL> del 4;

SQL> list;
  1  select customer_number
  2  , email,city
  3* from customer
```

Show the contents in the sql buffer

Delete the 4th line

Show the contents in the sql buffer

```
SQL> input where state='FL'

SQL> list;
  1  select customer_number
  2  , email,city
  3  from customer
  4* where state='FL'

SQL> clear buffer;
Buffer cleared
```

Add a line to the end of the current line in the buffer

Show the contents in the sql buffer

Erase the commands currently stored in the buffer

SQL*Plus Commands (cont'd)

- **SQL*Plus file commands**
  - ***SAVE, GET, START(****or @)***
    - » Save the contents of the SQL buffer into a script file
    - » Load a contents of script file into the SQL buffer
    - » Execute the contents of the specified script
  - ***SPOOL***
    - » Store query results in an operating system file
    - » SPOOL result.lst : start to write in result.lst
    - » SPOOL OFF : stop to write
  - ***HOST***
    - » Execute a host operating system command without leaving SQL*Plus
      - ex) HOST *dir* : execute a MS-DOS command *dir*

SQL*Plus Commands (cont'd)

- **SQL*Plus file commands**
  - ***SAVE, GET, START(***or ***@)***

```
SQL> select customer_number,
  2  email, city from customer
  3 where state='TX';
CUSTOMER_NUMBER      EMAIL                    CITY
---------------------------- ------------------------- ---------------
321654987            bfarmer@email.com        DALLAS
```

```
SQL> save query.sql;
```

Save buffer contents into a file

```
SQL> get query.sql;
  1  select customer_number,
  2  email, city from customer
  3  where state='TX'
```

Retrieve a file and place it into the buffer

```
SQL> @query.sql
CUSTOMER_NUMBER      EMAIL                    CITY
---------------------------- ------------------------- ---------------
321654987            bfarmer@email.com        DALLAS
```

SQL*Plus Commands (cont'd)

- **SQL*Plus file commands**
  - ***SPOOL, HOST***

```
SQL> spool result.lst
```
Start to write in result.lst

```
SQL> create table Spooled(name char(10));
Table created.
```

```
SQL> spool off
```
Stop to write

```
SQL> create table NotSpooled(name integer);
Table created.
```

```
SQL> spool result.lst append
```
Restart to write in result.lst

```
SQL> create table anotherSpooled(name char(10));
Table created.
```

```
SQL> spool off
```
Stop to write

```
SQL> host dir
result.lst  schema.sql insert.sql
```
Execute a command *ls*

```
SQL> edit result.lst
```
Open *result.lst* to see or edit

Assignment #1

Submission

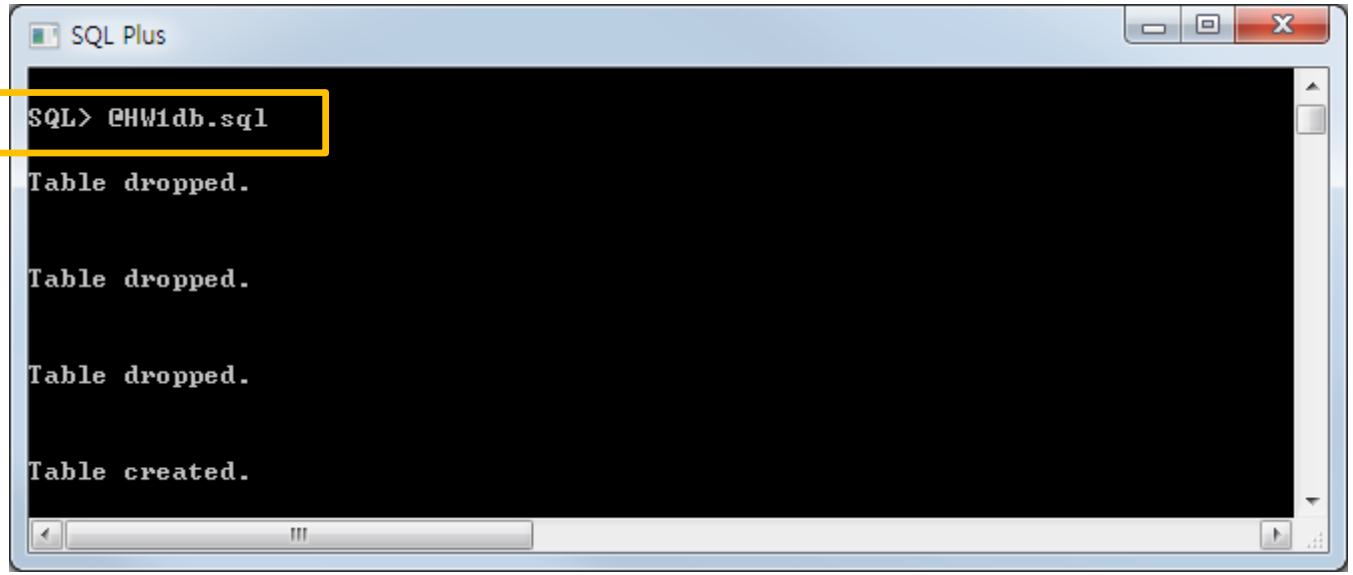
- **Due**
  - March. 23, 12 p.m.
  - Delay is not accepted
- **Submission standard**
  - *[student ID].lst* contains the executions of SQL commands and their results. You may use ***SPOOL*** command.
  - *[student ID].docx* includes relational algebra.
    For each given query, write it in relational algebra.
    - » Please write your student ID at first line of the file
  - Archive them into *[student ID].zip* and upload it to course homepage
- **Evaluation**
  - You will get points if your SQL queries have the assigned operators and find the right answers.
  - You will get points if your Relational algebra find the right answers.
  - Do not cheat others. Both of them will get no point.

Example Database

- **Create tables for assignment**

  1) Download *HW1db.sql* from the course homepage and Copy it to (directory that *Oracle Client* is installed)₩BIN

  2) ***@HW1db.sql*** or ***start HW1db.sql***



Example Database (cont'd)

- **Database Design**
  - You can see all the tables stored in your database using a command '**select * from tab**'

STUDENT	StudentID	Name	Department	Semesters	DoubleMajor
	2015610	Sofia	Bio and Brain Engineering	1	
	2012035	Amelia	Electrical Engineering	6	Computer Science
	...				

COURSE	CourseNo	Title	CourseType	Classroom
	CS360	Introduction to Database	Major Elective	E11, 402
	CS206	Data Structure	Major Required	N1, 102
	...			

TAKES	StudentID	CourseNo	SemTaken	ApplyDate
	2015610	CS360	'2015, Spring'	2015-02-10
	2012035	CS206	'2015, Spring'	2014-12-01
	...			

- *Semesters* : The number of semesters he/she enrolls
- *SemTaken* : The semester he/she took the course
- *ApplyDate* : The date he/she applied the course

Queries

- **Q1. Selection / Projection**
  - List the *StudentID* and *name* of the students whose *department* is 'Computer Science' in lexicographic order of their name.

- **Q2. Join**
  - List the *StudentIDs* of the students who take the courses whose classroom is 'E11, 402'. (without duplication)

Queries (cont'd)

- **Q3. Intersection (Use the predicate 'INTERSECT' not 'Join')**
  - List the *StudentID* of the students who enroll less than 4 semesters and take 'CS360'

- **Q4. Subquery (Use the set operator '(NOT) IN')**
  - List the *StudentID* and *department* of students who do not take any course.

Queries (cont'd)

- **Q5. Pattern matching**
  - For each student who takes any computer science department courses (*CourseNo* started with 'CS'), list the *StudentID* of the student and *CourseNo* of the CS courses he or she takes.

- **Q6. Subquery (Use the predicate 'ALL' or 'ANY')**
  - List the *name* of students in 'CS360' course where they applied latest (i.e. *ApplyDate* is the latest) among ALL students in the course.

Queries (cont'd)

- **Q7. Null values**
  - For students whose *DoubleMajor* field is not NULL, list their all the attributes of STUDENT tuples.

Relational Algebra Expressions

Write each query in relational algebra on [*student ID*].docx

- **Q-a**. List the *StudentID* and *name* of the students whose *department* is 'Computer Science' in lexicographic order of their name.
- **Q-b**. List once the *StudentIDs* of the students who take the courses whose classroom is 'E11, 402'.
- **Q-c**. List the *StudentID* of the students who enroll less than 4 semesters and take 'CS360'.
- **Q-d**. List the *StudentID* and *department* of students who do not take any course.
- **Q-e**. For each student who takes at least three courses, list their *StudentID* and the number of courses he or she takes .

References

- Lecture notes
- Text book
  - Chapter 6.1, 6.2, 6.3
- Oracle SQL Plus Tutorial
  - http://www.holowczak.com/oracle/sqlplus/